

## Aufgabenstellung:

Entwickelt ein Programm zur Verwaltung der an der Fachhochschule und PTL immatrikulierten Studierenden und Schüler/innen.

Hierfür sollen exemplarisch folgende Personendaten erfaßt werden:

Matrikelnummer (integer),

Name (string[30]),

Vorname (string[30]),

Anschrift :

    Straße mit HausNr. (string[30]),

    PLZ (string[10]),

    Wohnort (string[30]),

    Telefonnummer (string[30]),

    Email (string[30]),

Fachrichtung (Aufzählungstyp :(EA, IA, PA, AI, DI, II, MI, PI, WI)),

Verwaltungs- und Fachsemester (jeweils byte).

Erstellt ein *Menü* (Palette Standard: MainMenu) mit den Items

    Datei: Laden, Speichern, Programmende

    Person: Erfassen, Bearbeiten, Suchen

Grundsätzlich soll die Möglichkeit bestehen, innerhalb des Datenbestandes vor- und zurück zu blättern. Bei Auswahl des entsprechenden Menüpunktes soll ein neuer Datensatz erstellt, der angezeigte Datensatz geändert werden können. Zusätzlich kann der Benutzer einen bestimmten Datensatz durch Angabe der Matrikelnummer suchen lassen. *Ein Löschen ist nicht vorgesehen.*

Für die Matrikelnummer, die Fachrichtung, die PLZ, die Tel.-Nr. (Zahl, /, -) sowie das Verwaltungs- und Fachsemester (Zahl) ist eine Plausibilitätsprüfung durchzuführen. Benutzt z.B. die Komponenten MaskEdit, SpinEdit, RadioGroup. (Bedenkt hierbei auch, daß keine Matrikelnummer doppelt vergeben werden darf!)

## Hinweise:

- Es soll weder mit Arrays noch mit Zeigern noch mit Datenbankelementen gearbeitet werden, sondern mit Records einer typisierten Datei und Positionierung in derselben.
- Überlegt Euch aber sorgfältig die Unterteilung in Units und Prozeduren, damit in einer späteren Programmversion die direkte Dateiarbeit durch Zeigerarbeit ersetzt werden kann.  
Beispiel: Jetzt wird der eingegebene Record an eine Prozedur gegeben, die ihn an das Dateiende anfügt, nach einer Programmveränderung wird der Record an eine Prozedur gleichen Namens gegeben, die ihn in die Liste einfügt.
- Haltet Euch an die vorgegebene Recordstruktur, damit Ihr Euer Programm mit unseren Testdaten vorführen könnt.

## Tips:

- Benutzt für die Menüpunkte Dateiladen und -schließen den OpenFileDialog bzw. SaveDialog aus der Komponentenpalette Dialoge. Der OpenFileDialog kann dann im Ereignis DateiLadenOnClick aufgerufen werden mit  
`IF OpenFileDialog.Execute then //Datei gewählt`
- Schaut Euch die Eigenschaften des OpenFileDialog/SaveDialog an. Dort kann man schon viel voreinstellen (InitialDir, Filter).
- Im OpenFileDialog kann ein neuer Dateiname angegeben werden. Ihr müßt prüfen, ob `FileExists (OpenDialog.FileName)`.
- Ob eine gewählte Datei den falschen Typ hat, kann man etwas besser ausschließen, wenn man die Datei als `file of byte` öffnet, und dann prüft, ob ihre Größe ein Vielfaches der Recordgröße ist.  
`IF (filesize(f) mod sizeof(TPersonRecord)) = 0`
- Man kann jeden Menüpunkt `Enabled:=false` setzen, falls er noch nicht angewählt werden darf (Z.B. kein Datei schließen oder Personen bearbeiten, wenn keine Datei offen ist.)
- Hilfreich ist die Typwandlung  
`p.FB:= Tfachbereich(RGFB.Itemindex);`
- Zieht Ihr die Maus im Editor bei gedrückten <Strg><Alt> Tasten, so könnt Ihr einen Block markieren (probiert es mal aus).
- Denkt daran, nach jedem Datensatz `aus_der_Datei_lesen /_in_die_Datei_schreiben` den Dateizeiger wieder auf den Datensatz zu positionieren.